МИНИСТЕРСТВО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное   
образовательное учреждение   
«Благовещенский государственный педагогический университет»

Физико-математический факультет

# ОТЧЕТ

**по производственной практике**

Выполнил студент\_\_\_ курса \_\_\_\_\_ группы

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(ФИО)

Руководитель практики на предприятии

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(ФИО, должность)

Руководитель практики в БГПУ

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(ФИО, должность)

Оценка за практику \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Благовещенск

Издательство БГПУ

2018

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 4](#_Toc506546219)

[1 ОРГАНИЗАЦИОННАЯ СТРУКТУРА ПРЕДПРИЯТИЯ 5](#_Toc506546220)

[1.1 Общая характеристика учреждения 5](#_Toc506546221)

[2 АППАРАТНЫЕ И ПРОГРАММНЫЕ СРЕДСТВА 8](#_Toc506546222)

[2.1 Аппаратное обеспечение 8](#_Toc506546223)

[2.2 Программное обеспечение 8](#_Toc506546224)

[3 ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ 9](#_Toc506546225)

[3.1 Постановка задачи 9](#_Toc506546226)

[3.2 Этапы разработки 9](#_Toc506546227)

[3.3 Реализация проекта 9](#_Toc506546228)

[3.3.1 Разработка архитектуры 9](#_Toc506546229)

[3.3.2 Разработка функциональной части сайта(Бэк-энда) 11](#_Toc506546230)

[3.3.2.1 Доступ к веб-сайту 12](#_Toc506546231)

[3.3.2.2 Панель администратора 14](#_Toc506546232)

[3.3.2.3 Форма выгрузки отчетов. 15](#_Toc506546233)

[3.3.3 Разработка графической части(Фронт-энда) 17](#_Toc506546234)

[3.3.3.1 Окно авторизации 17](#_Toc506546235)

[3.3.3.2 Окно с ошибкой 18](#_Toc506546236)

[3.3.3.3 Окно администратора 18](#_Toc506546238)

[3.3.3.4 Окно с формой выгрузки отчетов. 20](#_Toc506546239)

[ЗАКЛЮЧЕНИЕ 23](#_Toc506546240)

ВВЕДЕНИЕ

Я, Либединский Сергей Александрович, проходил производственную практику по профилю специальности с 9 января 2018 г. по 19 февраля 2018 г. в Азиатско-Тихоокеанском банке на должности инженера-программиста. Во время которой были систематизированы приобретенные знания, а также получен опыт работы в учреждении.

Перед началом практики я ознакомилась с правилами внутреннего трудового распорядка, с правилами техники безопасности, со своими непосредственными обязанностями, основными функциями и задачами программиста. За время практики я ознакомилась с деятельностью учреждения, его документацией, структурой и задачами. Также меня ознакомили с профессиональными и должностными обязанностями остальных специалистов учреждения. Активно участвовала в деятельности отдела.

1. ОРГАНИЗАЦИОННАЯ СТРУКТУРА ПРЕДПРИЯТИЯ

1.1 Общая характеристика учреждения

Азиатско-Тихоокеанский Банк(АТБ) – имеет более чем 20-летнюю историю. До 2006 года в качестве «Амурпромстройбанка», после переименования и изменения формы собственности – «Азиатско-Тихоокеанский Банк» (ПАО).

АТБ – один из крупнейших региональных банков страны стабильно работает в 19 регионах Сибири и Дальнего Востока уже более 25 лет, постоянно входя в топ – 100 лучших банков страны по различным показателям, что подтверждено международными и российскими агентствами. Объем собственного капитала составляет 12,5 млрд рублей. По оценкам международных и российских рейтинговых агентств, банк демонстрирует уверенную стабильность. У АТБ 136,5 млрд рублей собственных активов, что позволяет качественно предоставлять услуги физическим лицам и корпоративным клиентам. Двести тринадцать отделений функционируют в 108 населенных пунктах- Чукотский и Еврейский автономные округи, Камчатский, Красноярский, Приморский, Хабаровский, Алтайский, Забайкальский края, Амурская, Иркутская, Сахалинская, Магаданская, Свердловская, Кемеровская области, республики Бурятия, Саха (Якутия) и Хакасия, а также Москва.

АТБ предлагает полный комплекс банковских продуктов и услуг – от потребительского кредитования до оплаты коммунальных платежей через терминалы АТБ и офисы банк.

Банк осуществляет следующие услуги:

* привлечение денежных средств физических и юридических лиц во вклады (до востребования и на определенный срок);
* размещение привлеченных денежных средств от своего имени и за свой счет;
* открытие и ведение банковских счетов физических и юридических лиц;
* осуществление расчетов по поручению физических и юридических лиц, в
* том числе банков-корреспондентов, по их банковским счетам;
* инкассация денежных средств, векселей, платежных и расчетных документов и кассовое обслуживание физических и юридических лиц;
* купля-продажа иностранной валюты в наличной и безналичной формах;
* привлечение во вклады и размещение драгоценных металлов;
* выдача банковских гарантий;
* осуществление переводов денежных средств по поручению физических лиц без открытия банковских счетов (за исключением почтовых переводов).
* Помимо перечисленных в пункте 3.1 раздела III Устава банковских операций Банк вправе осуществлять следующие сделки:
* выдачу поручительств за третьих лиц, предусматривающих исполнение обязательств в денежной форме;
* приобретение права требования от третьих лиц, уступку права требования третьим лицам и перевод долга на третьих лиц при исполнении обязательств в денежной форме, а также осуществление финансирования под уступку прав требования;
* доверительное управление денежными средствами и иным имуществом по договору с физическими и юридическими лицами;
* предоставление в аренду физическим и юридическим лицам специальных помещений или находящихся в них сейфов для хранения документов и ценностей;
* лизинговые операции;
* оказание консультационных и информационных услуг;

1. АППАРАТНЫЕ И ПРОГРАММНЫЕ СРЕДСТВА

2.1 Аппаратное обеспечение

В АТБ существуют кабинеты с рабочими местами, для сотрудников банка. Для каждого отдела свой кабинет. В кабинете, в который был назначен практикуемый, стояли 18 компьютеров с прикладными устройствами, для удобства работы.

2.2 Программное обеспечение

Свое рабочее место можешь настроить под себя, устанавливаешь необходимое программное обеспечение. В нашем варианте это: OS Windows 7 professional, Office 10 (Word, Excel, PowerPoint, Publisher), Oracle SQL Developer, Eclipse, Intellij idea, Notepad++, Netbeans и т.д.

1. ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ
   1. Постановка задачи

Необходимо разработать веб-сайт на платформе java для формирования и загрузки специальных отчетов из базы данных oracle, с правами доступа для сотрудников банка. Пользователю представляется возможность формировать отчеты за определённый срок и выбрать определенный тип. Администратор может регулировать – наименование типов, а также права пользователей. Доступ к веб-сайту будет существовать только внутри сети организации АТБ.

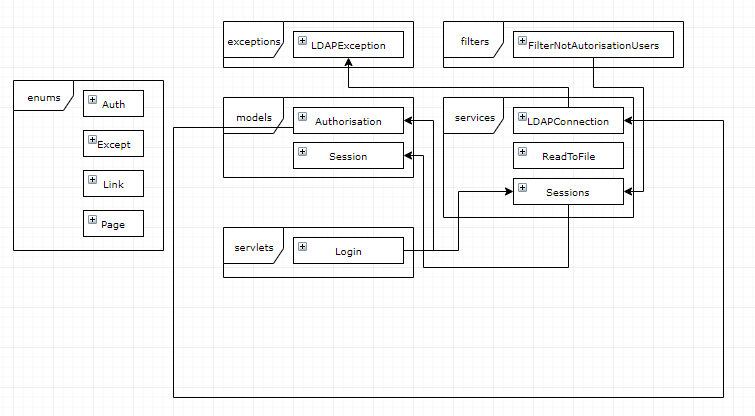
База данных уже существует, поэтому потребуется только выгрузка информации из нее, и упорядочение в файл.

* 1. Этапы разработки

1. *Разработка архитектуры*
2. *Разработка функциональной части сайта(Бэк-энда)*
3. *Разработка графической части(Фронт-энда)*
4. *Тестирование*
   1. Реализация проекта

3.3.1 Разработка архитектуры

Создания веб-сайта, было решено разделить на 4 модуля: модуль авторизации, главный модуль, модуль базы данных и модуль пользовательского интерфейса для веб-сайта.

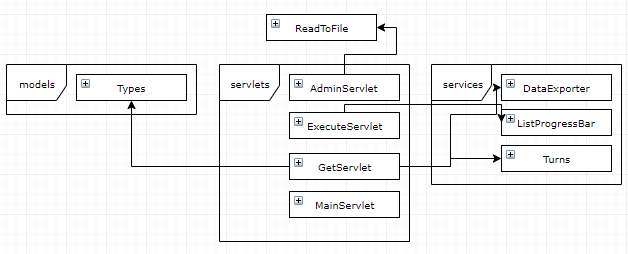
Рисунок 1 - Схема классов модуля авторизации

Модуль авторизации. На рисунке 1 присутствуют классы «Login» и «FilterNotAutorisation», они главные на этом рисунке, именно они будут отвечать за доступ к сайту. Сервлет (интерфейс Java, реализация которого расширяет функциональные возможности сервера) Login отвечает за отображение главной страницы авторизации, а «FilterNotAutorisationUsers» отвечает за проверку того что у пользователя есть доступ к ресурсу на который он пытается зайти. То есть прослушивает все ссылки на предмет несанкционированного доступа и отсеивает не авторизованных пользователей.

Элементы формы «enums», это статические поля которые используется во всей программе.

Элементы формы «exceptions» отвечает за исключительные ситуации, в классах.

В остальных формах создается единый функционал для полностью работающего разделения доступа.

Рисунок – Схема классов главного модуля

Главный модуль. На рисунке 2 в форме servlets изображено 4 сервлета отвечающие за главное отображение сайта. Остальные классы вспомогательные, для работы функционала сайта.

Модуль базы данных. Архитектура базы данных, скрыта из-за информационной безопасности банка. В вкратце можно охарактеризовать ее: она подключается к бд, запрашивает данные и формирует отчет из предоставленных данных.

Модуль пользовательского интерфейса для веб-сайта.



Рисунок – файлы со страницами jsp

На рисунке изображены все страницы, которые требуются для отображения веб-сайта.

3.3.2 Разработка функциональной части сайта(Бэк-энда)

Для отображения и работы требуются сервлеты, которые будут отвечать на GET и POST запросы.

Метод запроса POST предназначен для запроса, при котором [веб-сервер](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80) принимает данные, заключённые в тело сообщения, для хранения.

Метод запроса GET предназначен для запроса, при котором [веб-сервер](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80)  получает данные без заключения в тело сообщения.

Мы создали 5 сервлетов и 1 фильтр:

* Login – Сервлет который отвечает за авторизацию.
* FilterNotAutorithationUsers – Фильтр который отфильтровывает пользователей у которых нету доступа.
* AdminServlet – Сервелет отвечающий за окно с функциями администратора.
* ExecuteServlet – Вспомогательный сервлет созданный специально для прогресс бара.
* GetServlet – Вспомогательный сервлет, срабатывающий при нажатии кнопки «Получить» отчет.
* MainServlet – Сервлет отображающий форму выгрузки отчетов.

3.3.2.1 Доступ к веб-сайту

При переходе на любую страницу по домену веб-сайта, без авторизации или недостаточном уровне доступа вас будет отправлять на страницу авторизации. За это отвечает фильтр FilterNotAutorithationUsers(Рисунок 4).

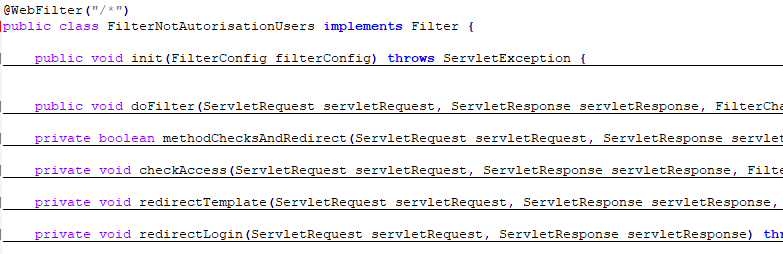


Рисунок – интерфейс класса FilterNotAutorisationUsers

При переходе на страницу авторизации будет отрабатывать сервлет Login, у которого существуют 2 метода doGet и doPost.

Метод doGet, удаляет авторизованных пользователей, которые зашли на страницу. Это служит как функция выхода. При нажатии кнопки выхода на других страницах, вас перенаправит, на окно авторизации, тем самым удалит из списка авторизированных пользователей.

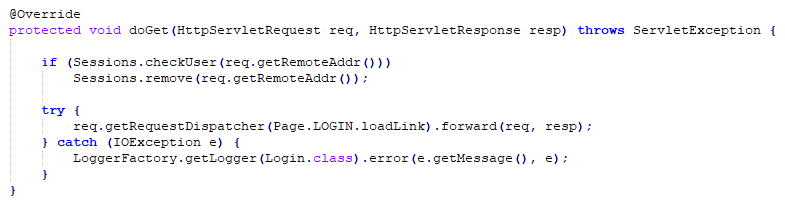


Рисунок – метод doGet класса Login

Как видно на рисунке 5 используется класс Sessions, именно он отвечает за пользователей. Если логин и пароль верен, то пользователь добавляется в иных случаях не добавляется или удаляется.

В метод doPost мы попадаем при нажатии кнопки «Войти». В параметры сохраняются логин и пароль, введенный в текстовые поля. От введенных данных зависит, куда вы будете перенаправлены. Три возможных случая: окно с ошибкой о неправильности логина и пароля, окно администратора или окно с возможностью выгрузки отчетов.

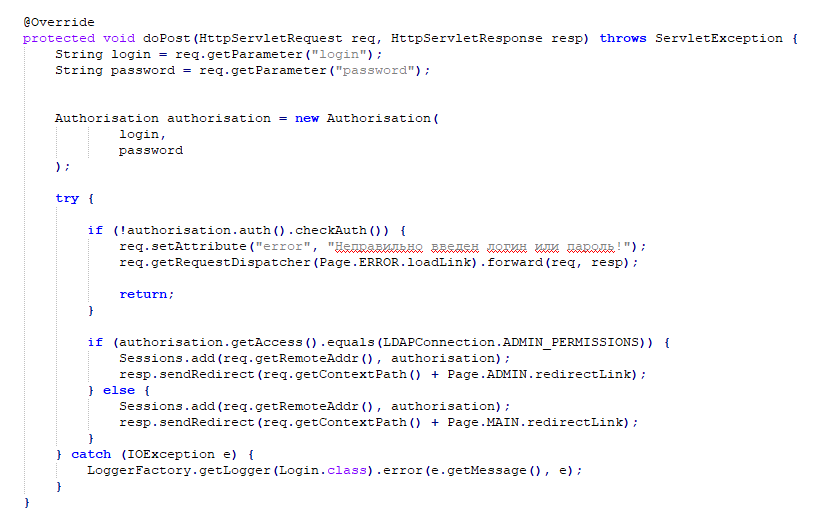


Рисунок – метод doPost класса Login

Класс Authorisation отвечает за проверку логина и пароля. База логинов и паролей хранится на сервере Tivoli(Система авторизированного доступа от компании IBM используемая в большинстве продуктов Java в банке). Обращаясь к серверу Tivoli, сервера возвращают параметры доступа. Получен он или нет, а также наименование прав который был выданы. Настройка Tivoli не осуществлялась, все права, и пользователи были уже созданы и настроены.

3.3.2.2 Панель администратора

Сервлет отвечающий за панель администратора – AdminServlet. На панель администратора можно попасть только с помощью логина и пароля единственного администратора. Который находится в конфигурационном файле.



Рисунок – интерфейс класса AdminServlet

В сервлете описаны следующие методы:

* doGet метод вызывается при переходе на панель администратора. И все необходимые поля для заполнения, заполняются данными который считываются c локальных файлов.
* doPost метод вызывается при взаимодействии с кнопками, кроме кнопки «Выход».
* saveProperties, вызывается из метода doPost. Сохраняет конфигурации заданные администратором.
* saveTypes, вызывается из метода doPost. Сохраняя изменённые названия типы данных.
* create, вызывается из метода doPost. Создает новые права для авторизации.
* delete, вызывается из метода doPost. Удаляет выбранные права для авторизации.
* save, вызывается из метода doPost. Сохраняет изменения в правах.

В основном сервлет работает с классом ReadToFile, который работает непосредственно с файлами конфигурации (сохранение, удалении и добавление).

3.3.2.3 Форма выгрузки отчетов.

За форму выгрузки отчетов, отвечают 3 сервлета: ExecuteServlet, GetServlet, MainServlet.

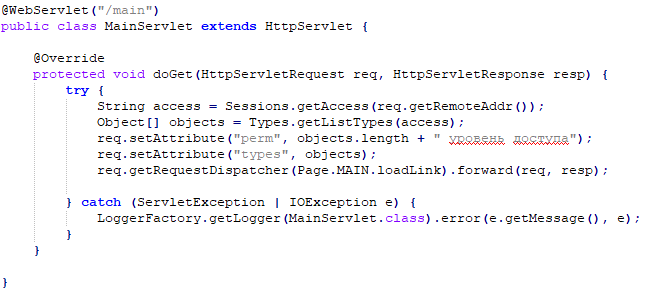


Рисунок – класс MainServlet

Сервлет отвечающий за загрузку страницы называется MainServlet. У сервлета всего один метод doPost, который отвечает за загрузку страницы и заполнения ее данными. В данном случае он загружает типы файлов, количество которых зависит от доступа, предоставляемого пользователю.

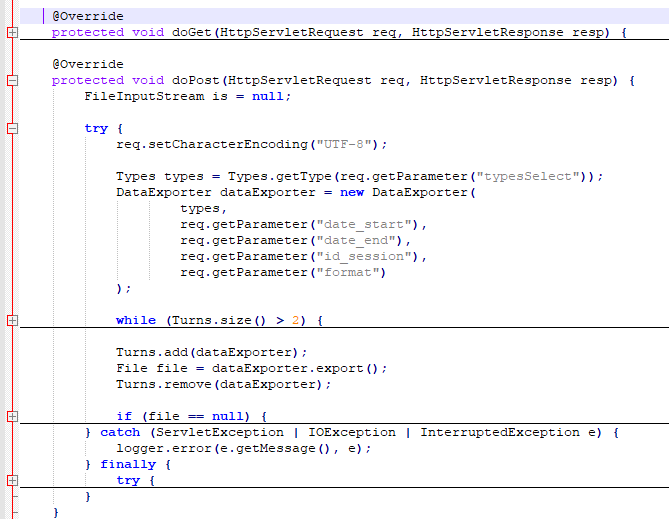


Рисунок – Класс GetServlet

В сервлете GetServlet, 2 метода: doGet и doPost. Сервлет вызывается в 2 случаях – если возникла ошибка скачки или отправлен запрос на скачку. Метод doGet отвечает за ошибку, а doPost за скачку.

В doPost методе – сначала загружается выбранные параметры, для которых нужно выгрузить отчет, после этого пользователь добавляется в очередь. Обработка файла начнется только после того как в очереди будет меньше 3 отчетов на обработку. Такое ограничение пришлось использовать, дабы не загружать базу данных.

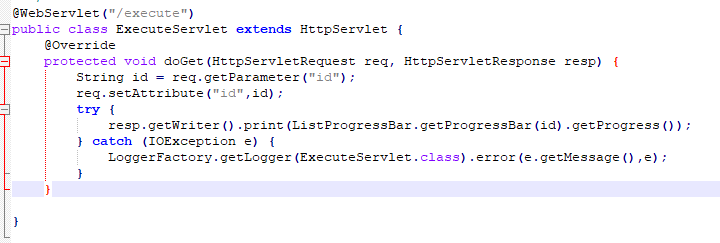


Рисунок – класс ExecuteServlet

ExecuteServlet отвечает за загрузку прогресс бара. В нем передается значение текущего значения прогресса. Как видно на рисунке 10 сервлет использует класс ListProgressBar именно он отвечает за все существующие прогресс бары.

3.3.3 Разработка графической части(Фронт-энда)

Для реализации, всех страниц, которые будут использовать в веб-сайте, потребовалось 4 jsp файла.

3.3.3.1 Окно авторизации

Страница авторизации состоит из формы с двумя текстовыми полями и кнопкой вход (Рисунок 10), также внизу есть сообщение об обращении в отдел разработчиков, для получения помощи при возникновении вопросов.

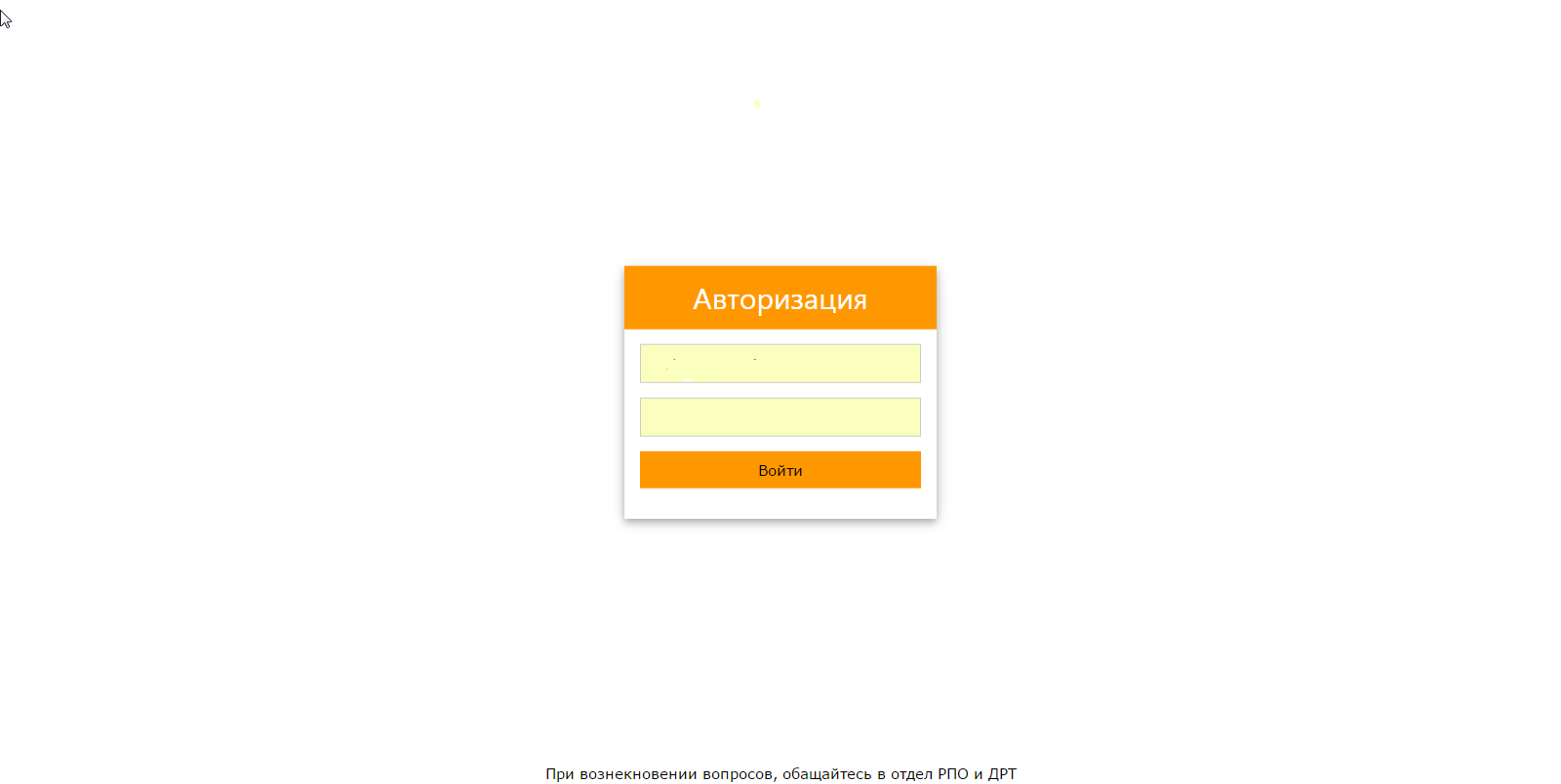


Рисунок – Окно авторизации

Как видно на рисунке 11 в форме описаны 2 поля и кнопка. В первой строке указано, стиль(class), метод отправки(method) и ссылка(action), куда требуется переадресовать запрос.

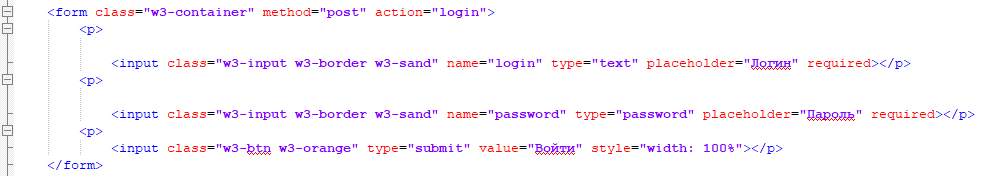


Рисунок - Фрагмент кода формы авторизации «login»

3.3.3.2 Окно с ошибкой

При возникновении исключительных ситуаций, в том числе и ввод неправильного пароля, появляется окно с ошибкой (Рисунок 12).

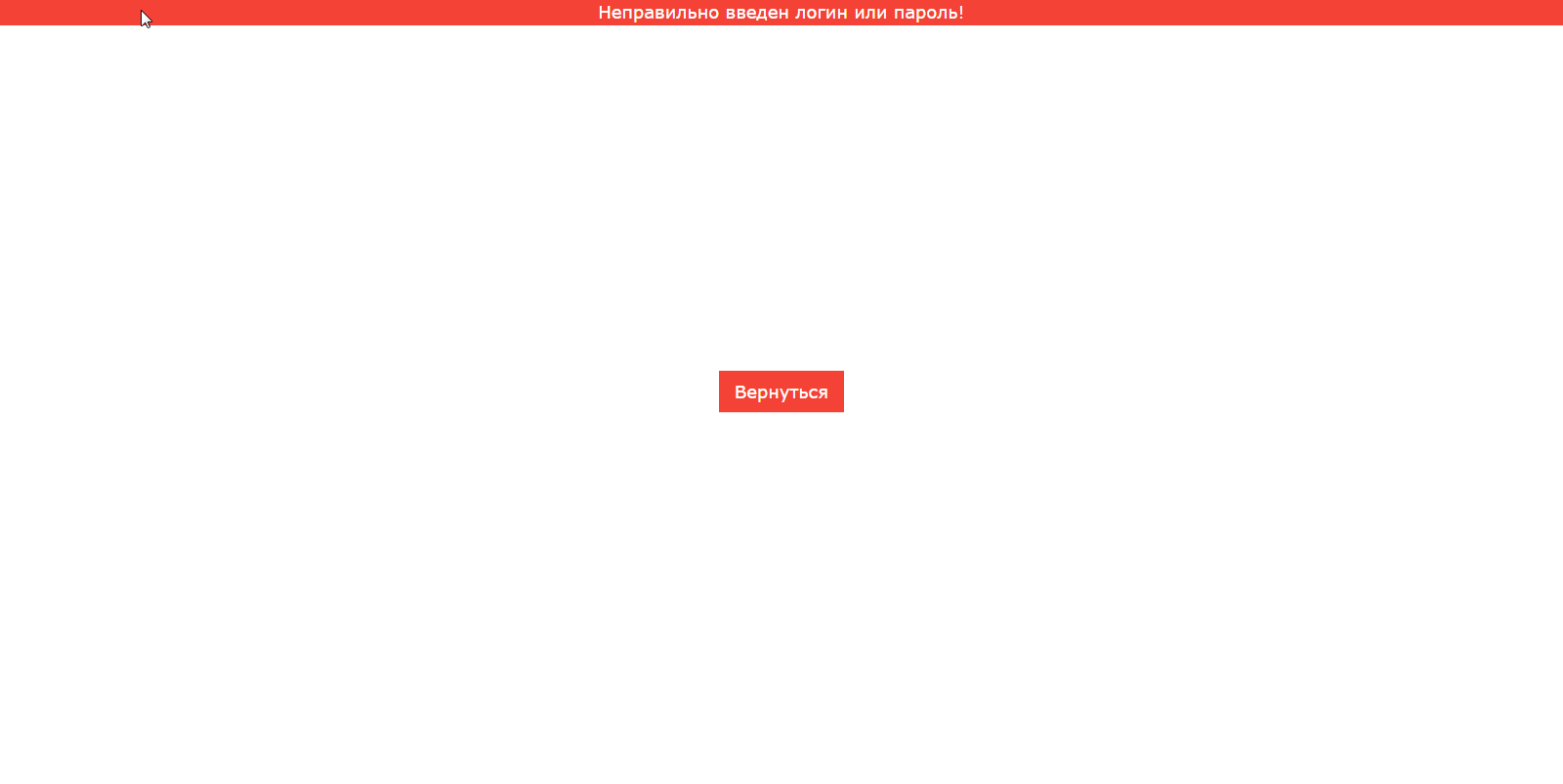


Рисунок – Окно с ошибкой

Была описана общая страница «error», на который передаётся параметр «error» с текстом ошибки.



Рисунок – Код страницы с ошибкой «error»

3.3.3.3 Окно администратора

На окне администратора отображаются: форма для создания прав, форма для просмотра прав, форма для редактирования прав, форма с редактированием конфигурационных значений и форма с редактированием названия типов (Рисунок 14), для соблюдения информационной безопасности, настоящие имена были скрыты.

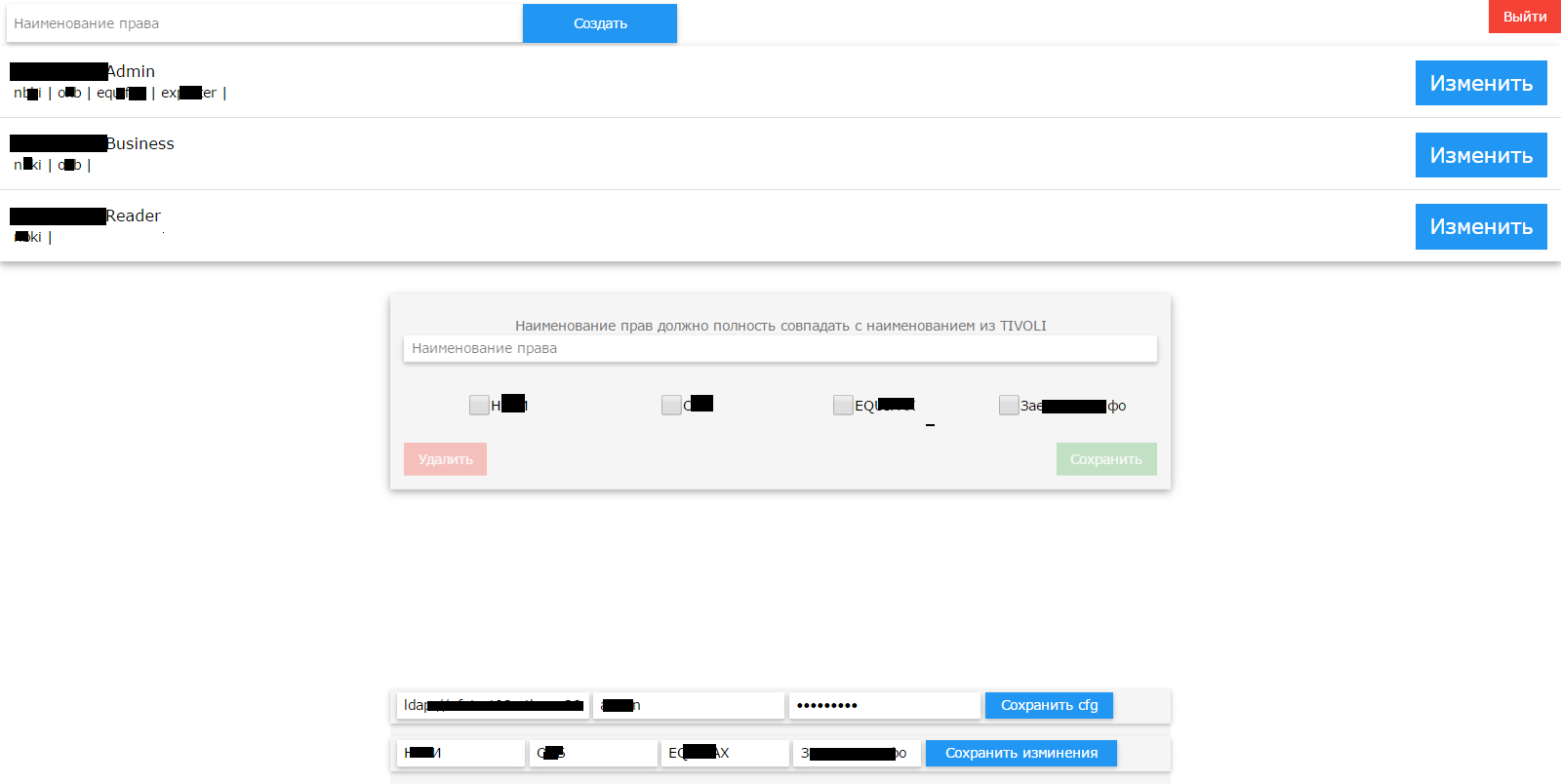


Рисунок – Окно с панелью администратора

Для перечисления всех возможных прав, в jsp файлах можно применять фрагменты java кода (Рисунок 15).

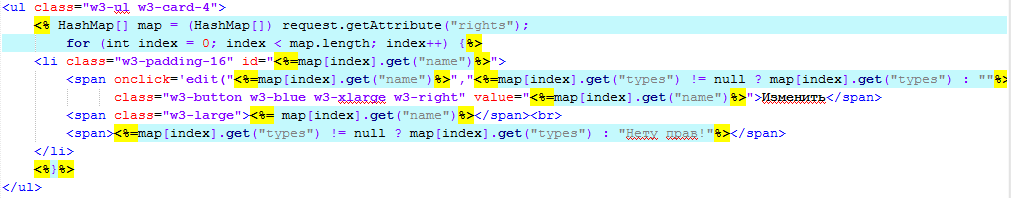


Рисунок – фрагмент кода панели администратора «admin», инициализация прав

Мы получает значения с атрибута «rights» и добавляем все значения, которые есть.

Также в админ панели потребовалось использовать функцию javascript (Рисунок 16) для редактирования типов. В целях информационной безопасности, настоящие наименования скрыты.



Рисунок – Фрагмент кода панели администратора «admin», функции «edit»

3.3.3.4 Окно с формой выгрузки отчетов.

После авторизации вы попадаете на окно с формой для выгрузки отчетов. Сверху слева, написан ваш уровень доступа (Обозначает, сколько типов выгрузки вам доступно). По центру форма с выбором типа, выбор даты выгрузки, формат файла и кнопка «Получить» (Рисунок 17).

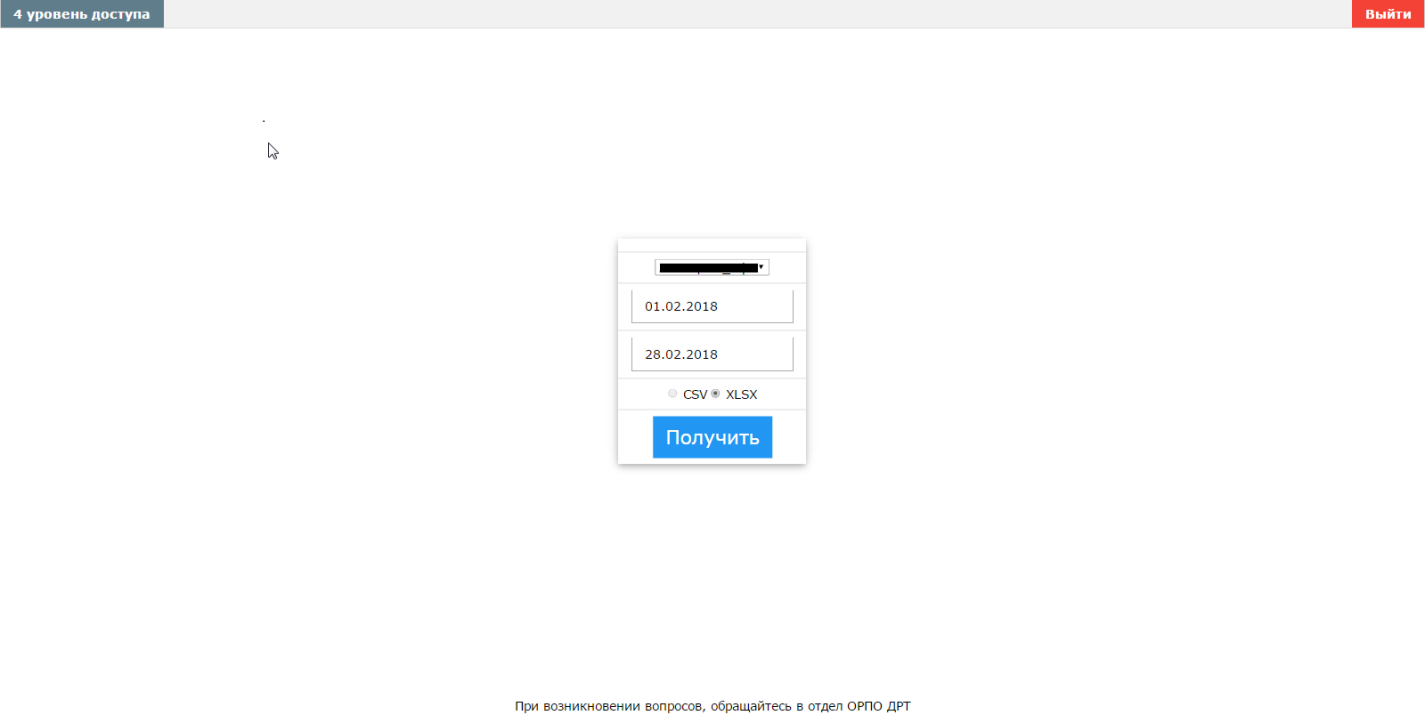


Рисунок – Окно загрузки отчетов

Загрузка типов в форму осуществляется с помощью цикла. На рисунке 18 видно как через цикл for запрашивая атрибут «types», мы добавляем все существующие типы для данного пользователя.

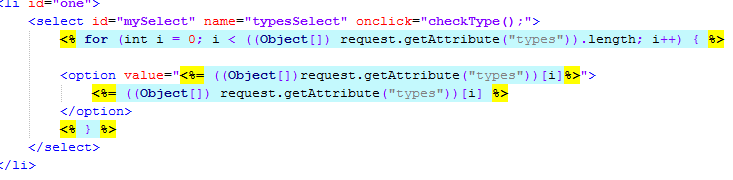


Рисунок – Фрагмент кода файла main, инициализация типов отчета.

После нажатия кнопки получить – возможны 3 варианта события.

* Выдаст ошибку, о том, что неверно выбраны поля.
* Начнется формирование (обработка) (Рисунок 19).
* Вы встанете в очередь на формирования отчета (Рисунок 20).

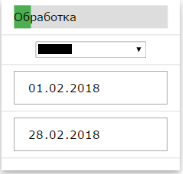


Рисунок – Форма с обработкой отчета

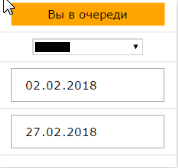


Рисунок – Форма с надписью «Вы в очереди»

После обработки отчета, начнется основной процесс. Он обозначается прогресс баром. После того как ползунок дойдет до 100%, файл автоматически начнет скачиваться на компьютер. А форма пропадет и появится кнопка «<Заново>»(Рисунок 21).

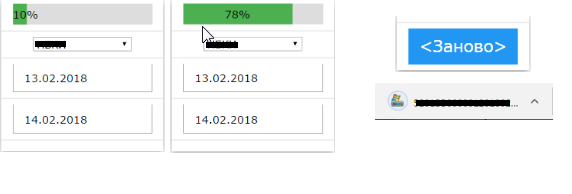
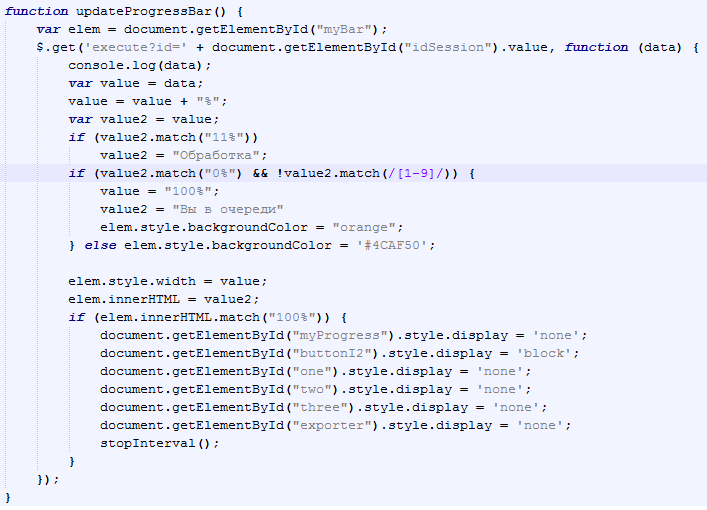


Рисунок – Процесс загрузки файла

За обновление прогресс бара отвечает javascript функция. При определенном значении цвет и текст прогресс бара меняется. Если значения прогресс бара равняется 100, то элементы формы скрываются, и появляется кнопка «<Заново>».

Рисунок – Фрагмент кода обновления прогресс бара в файле main

ЗАКЛЮЧЕНИЕ

В ходе практики были улучшены навыки программирования. Фиксированный план работ и установленные сроки его выполнения позволили научиться правильным образом организовывать свой труд. Так же поставленные цели и задачи были выполнены.